

7. A method as recited in claim 5 wherein a plurality of force effects are stored in said representation and in said device memory, wherein each of said force effects commanded to be output by said application program is summed to provide a total output force.

8. A method as recited in claim 1 wherein a greater number of said force effects can be stored in said representation than in said device memory.

9. A method as recited in claim 8 wherein when a particular one of said force effects that is stored in said representation and not stored in said device memory is commanded to be played by said application program, said particular force effect is sent to said device memory to replace a force effect stored in said device memory.

10. A method for managing the storage of force effects in a force feedback system, the force feedback system including a force feedback device connected to a host computer, the method comprising:

receiving a force effect create command by a driver running on said host computer, said command sent from an application program running on said host computer, said force effect create command instructing that particular force effect data for a particular force effect be stored in memory local to said force feedback device;

determining whether said local memory has sufficient space to store said particular force effect data;

if said local memory does have said sufficient space, sending said particular force effect data to said force feedback device to be stored in said local memory; and

if said local memory does not have said sufficient space, storing said particular force effect data in a cache implemented in memory of said host computer instead of said local memory.

11. A method as recited in claim 10 further comprising receiving a command by said driver from said application program to output said particular force effect to a user of said force feedback device, wherein if said particular force effect data is stored in said cache, said driver swaps said particular force effect data with loaded force effect data in said local memory and instructs said force feedback device to output said particular force effect.

12. A method as recited in claim 11 wherein said driver creates a representation of said local memory in said memory of said host computer.

13. A method as recited in claim 12 wherein said representation and said device memory each include an effect block and a parameter block, wherein an identifier of said particular force effect is stored in said effect block and at least one parameter for said particular force effect is stored in said parameter block.

14. A method as recited in claim 12 wherein said determining whether said local memory has sufficient space includes examining said representation for sufficient space.

15. A method as recited in claim 11 wherein said determining whether said local memory has sufficient space includes querying said force feedback device and receiving a response indicating whether sufficient space is available.

16. A method as recited in claim 12 wherein said determining whether said local memory has sufficient space includes comparing a priority of said particular force effect with a priority of said loaded force effect.

17. A method as recited in claim 16 wherein said priority of said of said particular force effect is compared to each priority of a plurality of force effects loaded in said device memory.

18. A method as recited in claim 16 wherein said priority of said loaded force effect is determined based at least partially on whether said loaded force effect is currently being output by said device.

19. A method as recited in claim 18 wherein said priority of said loaded force effect is determined based at least partially on the time period since said loaded force effect was last output by said device.

20. A method as recited in claim 16 wherein said priorities of said particular force effect and said loaded force effect are determined at least partially based on whether said loaded force effect is likely to be output based on a direction of movement of a manipulator of said force feedback device in a workspace of said manipulandum.

21. A method as recited in claim 20 wherein said likelihood of output of said particular and loaded force effects is also based on a velocity of said manipulandum of said force feedback device in said workspace.

22. A method as recited in claim 20 wherein said manipulandum controls a path of a cursor in a graphical user interface displayed by said host computer.

23. A method as recited in claim 16 wherein said priority of said loaded force effect is determined based at least partially on a predefined priority assigned to said loaded force effect.

24. A method as recited in claim 22 wherein said predefined priority was assigned based on a type of said force effect.

25. A method as recited in claim 22 wherein said predefined priority was assigned by said application program.

26. A method as recited in claim 10 wherein said force effect create command designates that at least one of a plurality of force effects be grouped in a category, and wherein said create command instructs that force effect data for said category of force effects be stored in memory local to said force feedback device in place of an existing category of loaded force effects.

27. A method as recited in claim 11 wherein when said local memory does not have sufficient space, said particular force effect is given a waiting status such that said force effect data for said particular force effect is sent to said device memory at a later time.

28. An apparatus for managing the storage of force effects in a force feedback system, the force feedback system including a force feedback device connected to a host computer, the method comprising:

means for receiving a force effect create command by a driver running on said host computer, said command sent from an application program running on said host computer, said force effect create command instructing that particular force effect data for a particular force effect be stored in memory local to said force feedback device;

means for determining whether said local memory has sufficient space to store said particular force effect data, wherein if said local memory does have said sufficient space, said particular force effect data is sent to said force feedback device to be stored in said local memory, and wherein if said local memory does not